FAA National Software Conference, May 2002 Development Tools Brainstorming

FAA Software Conference
Dallas/Ft.Worth, May 15, 2002

Software Development Tools Brainstorming Session

Andrew J. Kornecki, ERAU
andrew.kornecki@erau.edu
and
Leanna Rierson, FAA
leanna.rierson@faa.gov

Session Objective

- The session is designed to discuss the current state of the art and future in the area of software development tools as applied to development of safety-critical avionics software.

FAA National Software Conference, May 2002
Development Tools Brainstorming

Development Tools Research

- **Topic**: Assessment of Software Development Tools for Safety Critical Real-Time Systems
- **Funding**: the FAA Contract DTFA0301C00048
- **Duration**: Phase 1 (2002), Phase 2 (2003), Phase 3 (2004)
- **Team**: faculty and graduate students (ERAU, UCF)

Research Plan

- **Objective**:
  - to establish a base for assessment of software development tools
  - to create of a taxonomy and set of criteria/guidelines for the tool selection and qualification
- **Methodology**: searching the available literature and collecting information from the industry contacts, collecting practical tool(s) use data in experimental conditions to facilitate the assessment and tool evaluation.

FAA National Software Conference, May 2002
Development Tools Brainstorming

DO-178B Definitions

- ***Software tool*** *- A computer program used to help develop, test, analyse, produce or modify another program or its documentation.*
- ***Software development tools****: Tools whose output is part of airborne software and thus can introduce errors.*

Development Tools Variety

- **Tools to develop**
- **Tools to control**
- **Tools to produce**
- **Tools to build**
- **Tools to verify**
- **Tools to test**
- **Tools to load**

FAA National Software Conference, May 2002
Development Tools Brainstorming

Qualification (DO-178B)

- The objective of the tool qualification is to ensure that the tool provides confidence at least equivalent to that of the process(es) eliminated, reduced or automated
- A tool may be qualified only for use on a specific system …Use of the tool for other systems may need further qualification
- Only those functions that are used to eliminate, reduce, or automate software life cycle process activities, and whose outputs are not verified, need be qualified

Related Work Items

- Tool Categorization and Selection Criteria
- Tool Assessment Taxonomy
- List of Real-Time Development Tools
- List of Candidate Industry Entities Selected for Detailed Survey
- Tool Use Survey
- Preliminary Survey Results

FAA National Software Conference, May 2002
Development Tools Brainstorming

Three Questions (N8110.91)

- Can the tool insert an error into the airborne software or fail to detect an existing error in the software within the scope of its intended usage?
- Will the tool's output not be verified as specified in Section 6 of DO-178B?
- Are processes of DO-178B eliminated, reduced, or automated by the use of the tool? (i.e., will the output from the tool be used to either meet an objective or replace an objective of DO-178B, Annex A?

Details of Work

- Review of the existing tools used for real-time safety critical software development
- Creation of development tools taxonomy
- Creation of criteria for the tool selection
- Establishment of an experimental environment
- Collection of data on tool use
- Extraction of tool essential characteristics and establishment of guidelines for tool selection/qualification

FAA National Software Conference, May 2002
Development Tools Brainstorming

Taxonomy

- Taxonomy addresses the diversity of the software tools market.
- Taxonomy reveals numerous functionalities of software tools.
- Taxonomy provides the information allowing us to evaluate the functionalities.
- Taxonomy includes classification, which is of great explanatory value

Taxonomy Base

- **Characteristic features of real-time safety critical systems**: timeliness, responsiveness, schedulability, predictability, safety, reliability
- **Properties of the tools**: functionality and support specific development methodology, conformance to standards, modularity and scalability, speed and efficiency, support for the range of conceptual redundant design models, support for appropriate diagramming notations, traceability, consistency, vendor reputability, cost and availability, user community,

Development Tools Brainstorming

Tool attributes

- Functional
- Non-functional
  - Technical (dependability, performance, security)
  - Non-technical (vendors' reliability, support, cost, vendor certifications, training offered)
- Concerns - the parameters by which the attributes of a system are judged, specified and measured. Requirements are expressed in terms of concerns
- Attributes - properties of the system
- Evaluation methods - how we address the concerns

Development Tool Categories

- Software development tools categories, for safety critical application, as defined in the DO178B:
  - Requirements
  - Design
  - Coding
  - Integration

FAA National Software Conference, May 2002
Development Tools Brainstorming

Development Tool Categories: Requirements

- **Requirements** definition and analysis (including requirements modelling and verification)
- System **performance**, timing and sizing analysis (including simulation)
- Rapid **prototyping** (including algorithm analysis, man-machine interfaces analysis)
- Artifacts:
  - INPUT: system requirements (including safety), system architecture, hardware interfaces
  - OUTPUT: high-level software requirements, links to system safety assessment

Development Tool Categories: Design

- **Preliminary design** (software and interface requirements, traceability)
- **Detailed design** (analysis and decomposition, detailed software design, interface design)
- **Reuse** (reverse engineering, retargeting, code restructuring, source code translation)
- Artifacts:
  - INPUT: high-level software requirements
  - OUTPUT: low-level requirements, software architecture

FAA National Software Conference, May 2002
Development Tools Brainstorming

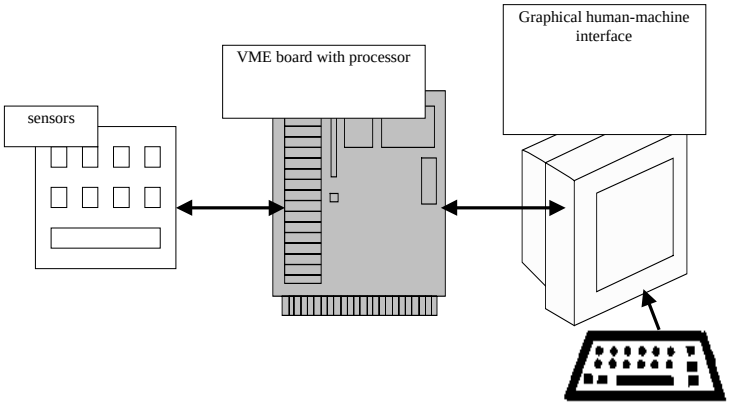
Development Tool Categories: Coding

- **Coding** (code generation)
- **Software System Build** (RTOS kernel customisation)
- Artifacts:
  - INPUT: low-level requirements, software architecture
  - OUTPUT: source code

Development Tool Categories: Integration

- **Build support** (compiler, linker, loader)
- **Run time support** (operating system, run time system)
- Artifacts:
  - INPUT: source code
  - OUTPUT: compiler directives, linker/loader commands, object code, executable code

Target Environment



Pilot Project: Development Steps

- Software installation
- Familiarization
- Design
- Use tools
- Analysis of results
- Implementation
- Integration
- Testing
- Report

FAA National Software Conference, May 2002
Development Tools Brainstorming

Topics to Discuss (1)

- Why do we need to qualify tools?
- What are the basic functionalities of a development tool?
- What are the categories of development tools?
- What tools were attempted to be qualified?
- How to achieve qualification for COTS tools?

Topics to Discuss (2)

- What kinds of development tools are you using today?
- What kinds of development tools do you anticipate using in the future?
- What tools are you considering for qualification?
- What are barriers to qualification of development tools

FAA National Software Conference, May 2002
Development Tools Brainstorming

Topics to Discuss (3)

- How do object-oriented tools fit into the picture? (do they change our paradigm?)
- What issues need to be addressed regarding development tools and qualification?
- What would help to encourage safe usage of development tools?
- What would help to enable tool qualification?

Issues

- ❑ Certified systems data-base
- ❑ Certification: U.S. and Europe
- ❑ Qualification acceptance and reusability
- ❑ MBV Tools

Development Tools Brainstorming

Clarification Needed (1)

- Are there any other artifacts produced by the four development processes (phases) we should consider in the analysis?
- What tools allow the developer to automate the transformation from the input to output artifacts of the above phases?
- Are there currently any tools on the market that support such picture of transition?

Clarification Needed (2)

- When qualifying a tool, do we consider the tool as a black box or white box?
- When qualifying a tool, co we consider such tool attributes as e.g. vendor reputability and history, availability of support, quality and completeness of documentation, etc.?
- Did we ever attempted/considered tool qualification not in the scope of specific application?

FAA National Software Conference, May 2002
Development Tools Brainstorming

Clarification Needed (3)

- We had in the past effort to “validate” compilers (Ada compilers are the case in point) - is validation an equivalent of generic qualification?
- Is a board-support-package (BSP) treated as one of the components of the application to be certified?
- Are external libraries and their integration into the application software a concern to qualification? What is the relation of libraries and BSP to the integration tools?

Tools selected for detailed evaluation

- Design Analysis and Decomposition:
  - Rose Real Time from Rational
  - Rhapsody from iLogix
  - Esterel Studio and SCADE from Esterel Technologies
  - StateMate from iLogix
  - Artisan Studio from Artisan
- RTOS Kernel Customisation tools:
  - Tornado from Wind River Systems
  - Multi IDE from Green Hills Software
  - OSE IDE from OSE Systems